

Object-Oriented Programming Language

10/23/2014

Homework Assignment No. 4

Due 09:00 pm, Wednesday October 29, 2014

Late submission within 24 hours: score*0.9;

Late submission before post of solution: score*0.8 (the solution will usually be posted within a week); no late submission after the post of solution)

(Total 50%)

1. (30%) C++ does not provide a built-in way of representing or manipulating the mathematical polynomial such as $6x^3-5x+10.2$. One way to represent the polynomial is to use `vector<double>` to store the coefficients in which the index of the vector is the exponents of the corresponding terms and the element of the vector is the coefficient. A home-made input syntax via `string` allows us to parse the input. For example, a very simple home-made string representation of $6x^3-5x+10.2$ is:

```
| 10.2 x^0 -5 x^1 0 x^2 6 x^3
```

And the contents of the `vector<double>` for the polynomial representation are:

```
| [10.2, -5.0, 0.0, 6.0]
```

Write a Polynomial class that provides a constructor and add and print member functions. Put your class definition in a header file `poly.h` and implement the constructor and member functions in a cpp file `poly.cpp`. Test your program with the client code `Prob3.cpp` listed below and given in the `FirstMidtermMock.zip`:

```
| #include "Polynomial.h"
| #include <string>
| #include <iostream>
| using namespace std;
|
| int main ()
| {
|     cout << "Please enter the first polynomial with a home-made string
representation: " << endl ;
|     string poly_rep ;
|     getline(cin, poly_rep) ;
|     Polynomial poly1(poly_rep) ;
|
|     cout << "Please enter the second polynomial with a home-made string
representation: " << endl ;
|     getline(cin, poly_rep) ;
|     Polynomial poly2(poly_rep) ;
```

Object-Oriented Programming Language

10/23/2014

```
    cout << "The first polynomial after the addition of the second  
polynomial is: " << endl;  
    poly1.combine(poly2).print(cout) << endl ;  
  
    return 0 ;  
}
```

Two sample runs look like:

```
Please enter the first polynomial with a home-made string representation :  
1.2 x^0 3 x^1 0 x^2 3.5 x^3  
Please enter the second polynomial with a home-made string representation :  
0.3 x^0 -2.2 x^1 1.2 x^2  
The first polynomial after the addition of the second polynomial is(print) :  
1.5 x^0 0.8 x^1 1.2 x^2 3.5 x^3  
請按任意鍵繼續 . . .
```

```
Please enter the first polynomial with a home-made string representation :  
1.2 x^0 2.3 x^1  
Please enter the second polynomial with a home-made string representation :  
3 x^0 2.1 x^1 1.2 x^2 0 x^3 3.2 x^4  
The first polynomial after the addition of the second polynomial is(print) :  
4.2 x^0 4.4 x^1 1.2 x^2 0 x^3 3.2 x^4  
請按任意鍵繼續 . . . ■
```

2. (20%) Now release the format constraint in Problem 3 so you can parse a polynomial $6x^3-5x+10.2$ represented by:

```
| 6 x^3 -5 x^1 10.2 x^0
```

Or

```
| -5 x^1 10.2 x^0 6 x^3
```

Test your implementation using the same client code given above (Prob3.cpp) but enter the first and second polynomials that support the new polynomial representations. Below is a sample run.

```
Please enter the first polynomial with a home-made string representation :  
6 x^3 -5 x^1 10.2 x^0  
Please enter the second polynomial with a home-made string representation :  
-5 x^1 10.2 x^0 6 x^3  
The first polynomial after the addition of the second polynomial is(print) :  
20.4 x^0 -10 x^1 0 x^2 12 x^3  
請按任意鍵繼續 . . .
```







HW Submission Procedure (請仔細閱讀):

Object-Oriented Programming Language

10/23/2014

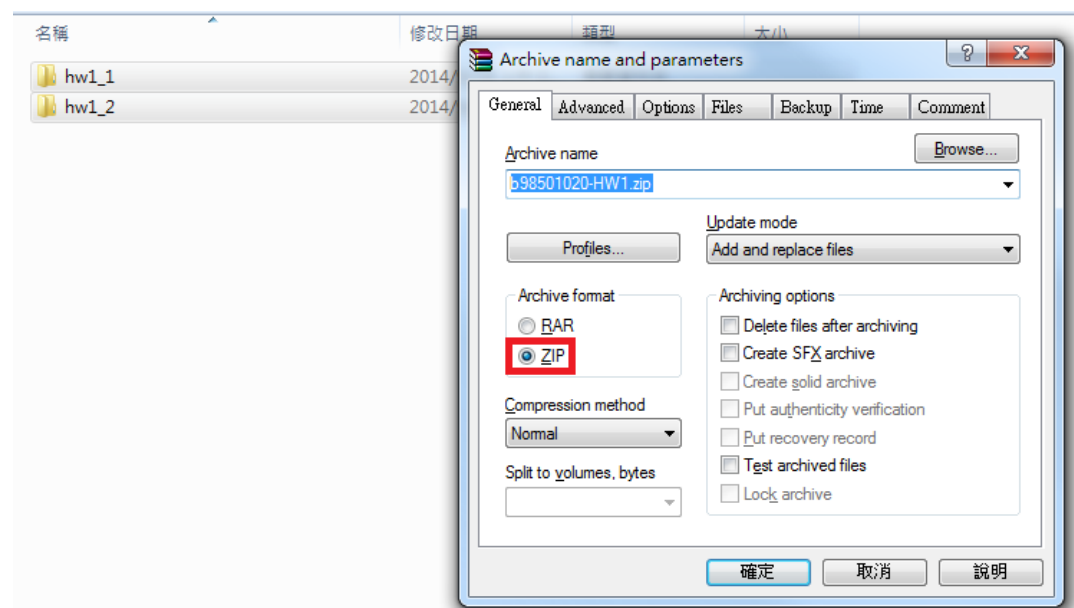
1. You should create an independent Visual Studio 2013 project for each problem. You should submit your project folder, which includes your source codes (header and cpp files), input/output data and Visual Studio 2013 project files. To save space, please delete Debug folder and sdf before submitting your files.

繳交時請以專案資料夾區分每一道題目，資料夾中應包含該題中的每個標頭檔、cpp 檔、所讀取的資料與輸出的結果與 Visual Studio 2013 相關專案檔。繳交前請刪除 Debug 資料夾與 sdf 檔案，以節省空間。

	Debug	2014/9/18 上午 0...	檔案資料夾	
	HW1-Prob1.sdf	2014/9/18 上午 0...	SQL Server Conn...	64 KB
	HW1-Prob1.sln	2014/9/18 上午 0...	Microsoft Visual ...	1 KB
	HW1-Prob1.vcxproj	2014/9/18 上午 0...	VC++ Project	4 KB
	HW1-Prob1.vcxproj.filters	2014/9/18 上午 0...	VC++ Project Filt...	1 KB
	Prob1.cpp	2014/9/17 下午 0...	C++ Source	1 KB

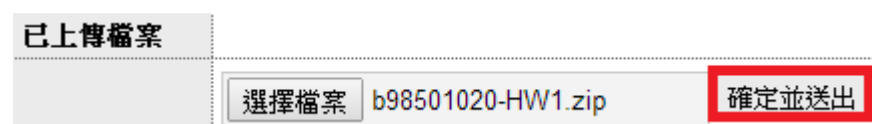
2. You should zip all the files and use your student id to name the zip file (e.g., b98501020-HW1.zip).

將所有檔案及資料夾收納在以學號命名的 zip 壓縮檔中。(例如：b98501020-HW1.zip)



3. Submit your HW directly through the course website.

請直接透過課程網站繳交作業。



HW Grading Policy:

1. You should consider about exception handling, e.g. error input, file opening fail, etc.
請注意所有例外狀況的處理，例如：錯誤的符號字串輸入、檔案開啟失敗等。
2. The coding style includes your output format.

輸出資料的格式將納入格式評分。

3. **If your code is not compilable, your score in this problem is zero (including coding style).**
若程式無法編譯，則該題以零分計算。(包含格式分數)
4. Your program will be tested with other data which is not the same as provided samples.
除了題目所提供的範例測試資料以外，作業程式碼將以額外的測試資料進行測試。
5. If tricky situations occur, the grade depends on Prof. Chen or TA's judgment.
假如有特殊情況發生，則依據陳俊杉教授以及助教們的判斷給分。

- Coding Style (20%): 編碼格式分數

1. format

整體形式與輸出資料的格式

2. readability

可讀性

3. variables naming

變數命名方式

4. typesetting

型別設定

- Functionality (80%): 功能性分數

1. run-time performance:

執行時的表現

1) samples not passed -> x

範例測資錯誤 => 此部分零分

2) samples passed but some tests failed -> partial

範例測資通過但是部分測資失敗 => 部份給分

3) samples and tests all passed

範例測資與所有測資通過 => 此部分滿分